Photometric Attitude Estimation using Gaussian Process Regression

Ryui Hara, Yasuhiro Yoshimura, Toshiya Hanada
Department of Aeronautics and Astronautics, Kyushu University

ABSTRACT

The rapid growth of resident space objects (RSOs) in Earth's orbit has intensified the need for advanced space situational awareness (SSA) (or space domain awareness, SDA) to manage satellite traffic and prevent collisions. Attitude estimation is critical for accurate state propagation, as non-gravitational forces like solar radiation pressure and atmospheric drag depend on the object's attitude. This study explores using light curves—brightness data from ground-based telescopes—to estimate RSO attitudes. Light curve inversion, traditionally used in astronomy, faces challenges when applied to RSOs due to their non-convex shapes and anisotropic reflections. Conventional methods for attitude estimation often assume known shape and surface parameters, which are usually unknown for space debris generated by a collision or breakup. To address this, this study proposes the estimation method with GPUKF, combining Gaussian process regression (GPR) with the unscented Kalman filter (UKF). This study uses GPR for a non-parametric observation model, enhancing robustness against unknown surface parameters. This study applies GPUKF to a Box-Wing object on a geosynchronous orbit. Numerical simulation results demonstrate that GPUKF works better than a conventional UKF and provides robust attitude estimation against surface property variations, contributing to practical scenarios in SSA/SDA where the object parameters are unknown.

1. INTRODUCTION

Recent advances in large constellations drastically increase the population of resident space objects (RSOs) around the Earth [1]. Moreover, once a collision or breakup of satellites occurs, a lot of space debris is generated. Thus, space situational awareness (SSA) (or space domain awareness, SDA) aiming to detect, track, and understand the motion of Earth-orbiting RSOs, is becoming more important in managing the traffic of satellites. For highly accurate SSA/SDA, much information such as orbit, attitude, and shape of RSOs enables improving the accuracy of state propagation of RSOs. The attitude of RSOs is especially a key state for improvement because non-gravitational forces such as solar radiation pressure (SRP) and atmospheric drag depend on the object's attitude. In this context, this study focuses on the attitude estimation of an RSO via light curves.

Light curves are a series of space objects' brightness observed by ground-based telescopes. The light curves depend on the orbit, attitude, shape, and surface properties of the space object. In other words, these characteristics can be inversely deduced from light curves. This is known as light curve inversion and is a promising and cost-effective method to estimate the RSO's attitude. Light curve inversion has been studied for a long time in the field of astronomy to determine rotation periods, rotation axis, and shape of asteroids [2, 3]. Although asteroids are usually assumed to have a convex shape and diffusive reflection, the RSOs have non-convex shape and optical properties including anisotropic reflection, which significantly makes the light curve inversion of the RSOs challenging. In previous studies, various attitude estimation methods via light curves have been proposed. For example, Fabrizio et al. [4] determine an RSO's attitude by using the 3-D virtual reality model. Piergentili et al. [5] describe the attitude estimation method using the unscented Kalman filter (UKF). Matsushita et al. [6] achieve highly accurate attitude estimation by the use of the sudden change of light curves, called glint. There are various research about Earth-orbiting objects' attitude estimation via light curves. However, most of the proposed methods assume that the target object's parameters such as shape and surface parameters are known, though they are usually unknown for space debris generated by a collision or breakup. When the shape or surface parameters of the object are unknown, the light curves cannot be calculated because they are parametrically formulated with the object's orbit, attitude, shape, and surface parameters.

This study tackles this problem by combining the Gaussian process regression (GPR) and the UKF, called GPUKF. GPR is one of the machine learning methods that allows non-parametric regression [7]. The UKF is one of the Bayesian filters and is used for filtering the noise and estimating the state of non-linear systems from observed values. A conventional UKF consists of a prediction step and an observation step in the estimation sequence, which require parametric models. On the other hand, the GPUKF can represent the models non-parametrically by using the GPR.

This study uses the GPR to describe the light curve model in the observation steps of the estimation sequence non-parametrically. Thus, the object attitude can be estimated from the light curves with GPUKF even if the shape or the surface parameter are unknown. Furthermore, thanks to the non-parametric model, this estimation method is robust against those parameters.

Although the GPR would enable describing the system model and the light curve model in the prediction step and the observation step, respectively, this study focuses on the GPR of the light curve model. To this end, the inputs for the training data are set to the attitude angles covered entirely and the mean of the past light curves for a certain period of time. The outputs are set to the light curves corresponding to the input attitude angles. In this study, the target object is on a geosynchronous orbit, and the time of period of the observation is several minutes. Hence, the relative position of the object–observer–Sun is fixed for simplicity. In addition, a Box-Wing object is used for both the training and validation. To verify the GPUKF for the attitude estimation via light curves, the validation data are calculated by propagating the equation of Euler's rotation motion and calculating the light curve model.

The organization of this paper proceeds as follows. First, the necessary formulae for numerical simulation are reviewed, such as BRDF, light curve model, attitude representation, GPR, and UKF in Section 2. Second, the application of the GPR with the UKF to attitude estimation is described in Section 3. Then, the configuration of the training data and validation data are described. Finally, Section 4 shows the attitude estimation results compared with a conventional UKF. The comparison shows that the GPUKF enables the attitude estimation to have robustness against surface properties, contributing to practical scenarios in SSA/SDA where object parameters are unknown.

2. PRELIMINARIES

2.1 Bidirectional Reflectance Distribution Function (BRDF)

A BRDF f_r is the function that describes the objects' reflection property [8]. A BRDF is represented as

$$f_r(\boldsymbol{x}, \boldsymbol{s}, \boldsymbol{v}) = \frac{L_r(\boldsymbol{x}, \boldsymbol{v})}{L_i(\boldsymbol{x}, \boldsymbol{s})} \tag{1}$$

where L_r is reflected radiance, L_i is irradiance, $\boldsymbol{x}$ is the position vector of the target facet, $\boldsymbol{s}$ is the light source direction unit vector, and $\boldsymbol{v}$ is the observer direction unit vector. The geometry of the reflected light on a facet is illustrated in Fig. 1 where $\boldsymbol{n}$ is the normal unit vector of the facet. The vectors $\boldsymbol{s}$ and $\boldsymbol{v}$ are defined with the sets of azimuth and polar angles, (ϕ_i, θ_i) and (ϕ_r, θ_r), respectively, and they define the bisector vector $\boldsymbol{h}$ as

$$\boldsymbol{h} = \frac{\boldsymbol{s}+\boldsymbol{v}}{\|\boldsymbol{s}+\boldsymbol{v}\|} = \begin{bmatrix} \sin\alpha\cos\beta \\ \sin\alpha\sin\beta \\ \cos\alpha \end{bmatrix} \tag{2}$$

The BRDF is also represented as

$$f_r = dc_d + sc_s \tag{3}$$

where d is the fraction of the diffusion and s is the fraction of the specularity. In this study, the Ashikhmin-Shirley model [9] is used. The Ashikhmin–Shirley model is anisotropic model and written as

$$c_d = \frac{28\rho}{23\pi}(1 - sF_0)\left[1 - \left(1 - \frac{\boldsymbol{n}^T\boldsymbol{s}}{2}\right)^5\right]\left[1 - \left(1 - \frac{\boldsymbol{n}^T\boldsymbol{v}}{1}\right)^5\right] \tag{4}$$

$$c_s = \frac{\sqrt{(n_u+1)(n_v+1)}}{8\pi}\frac{F}{\boldsymbol{v}^T\boldsymbol{h}\max\{\boldsymbol{n}^T\boldsymbol{s}, \boldsymbol{n}^T\boldsymbol{v}\}}(\cos\alpha)^\gamma \tag{5}$$

where ρ is the factor that specifies the diffuse reflectance of the substrate under the specular coating and n_u and n_v are exponential factors that define the shape of the specular lobe. The exponent γ and the Fresnel reflectance F are defined as

$$\gamma = n_u\cos^2\beta + n_v\sin^2\beta \tag{6}$$

$$F = F_0 + \left(\frac{1}{s} - F_0\right)\left(1 - \boldsymbol{v}^T\boldsymbol{h}\right)^5 \tag{7}$$

where F_0 is the material's reflectance for the normal incidence.

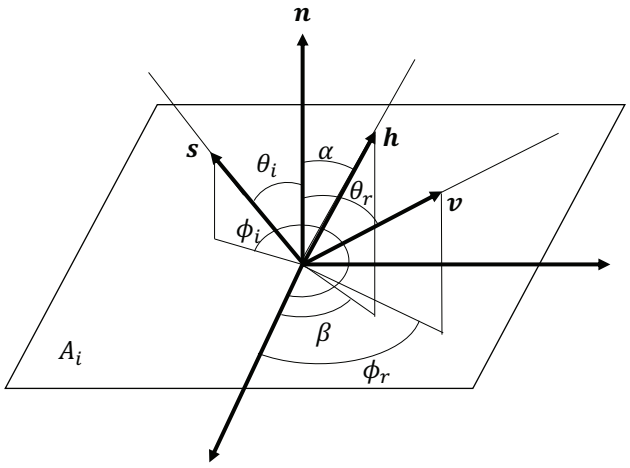


Fig. 1: Light reflection on a facet

2.2 Light Curve Model

Light curves are a series of the space objects' brightness observed by a telescope and their intensity depends on incoming light and reflected light. The relative magnitude of the brightness of a facet observed on the ground is described as follows.

$$m_{\text{app}} = m_{\text{sun}} - 2.5 \log_{10} \frac{f_{\text{obs}}}{r_{\text{obs}}^2} \tag{8}$$

where $m_{\text{sun}} = -26.7$ is the magnitude of the brightness of the Sun and r_{obs} is the distance between the facet and the observer. It is assumed that r_{obs} is the same for all the facets because the distance is much longer than the size of the object. The magnitude term f_{obs} is written as

$$f_{\text{obs}} = f_r f_i A(\boldsymbol{n}^T \boldsymbol{v}) \tag{9}$$

where A is the area of the facet and f_i expresses the fraction of the sunlight with respect to the normal unit vector of the facet as

$$f_i = \begin{cases} 0 & (\boldsymbol{n}^T \boldsymbol{s} \leq 0) \\ \boldsymbol{n}^T \boldsymbol{s} & (\boldsymbol{n}^T \boldsymbol{s} > 0) \end{cases} \tag{10}$$

2.3 Kinematics of a Space Object

The coordinate system and the frame used in this study are shown in Fig. 2. The XYZ coordinate system is used to represent the position of the space object orbiting the Earth, the observer, and the Sun. The XYZ system originates at the center of mass of the space object and the fundamental plane is the Earth's equator. The X axis points in the direction of the observer as a preferred direction. The Y and Z axes consist of the right-handed frame. This coordinate system is an inertial system because the relative position of the object–observer–Sun is fixed for simplicity in this study. The body-fixed frame in Fig. 2 is used to represent the space object attitude. It originates at the center of mass of the space object. Each axis $\boldsymbol{x}_b$, $\boldsymbol{y}_b$, and $\boldsymbol{z}_b$ points towards the direction of the principal axes of inertia.

This paper uses quaternions to represent an attitude with four parameters that include unit rotation vector (3 degrees of freedom) and rotation angle (1 degree of freedom) from Euler's theorem for rigid body rotation. A quaternion is defined as

$$\begin{aligned} \boldsymbol{q} &= [q_1, q_2, q_3, q_4]^T = [\bar{\boldsymbol{q}}^T, q_4]^T \\ &= \begin{bmatrix} \boldsymbol{e} \sin \frac{\phi}{2} \\ \cos \frac{\phi}{2} \end{bmatrix} \end{aligned} \tag{11}$$

where $\boldsymbol{e}$ is a unit rotation vector, ϕ is a rotation angle, and $\bar{\boldsymbol{q}}$ and q_4 are called vector part and scalar part of the quaternion, respectively. The advantage of using quaternions as the attitude representation is that quaternions have no

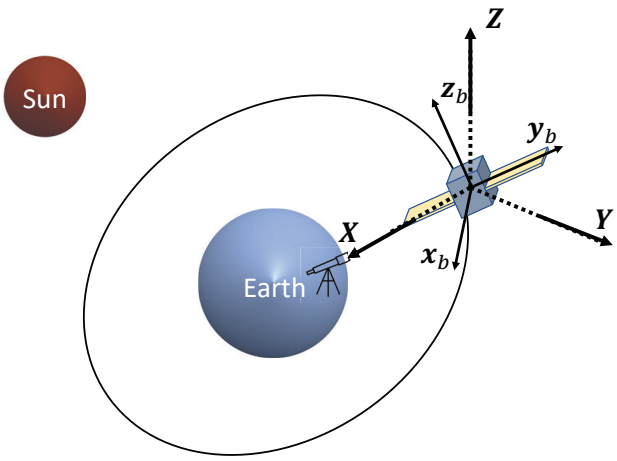


Fig. 2: XYZ coordinate system and body-fixed frame

singularities. However, it has the constraint that its norm is one as shown in Eq. (12), and the calculation for quaternions cannot be applied to four arithmetic operations as is.

$$\boldsymbol{q}^T \boldsymbol{q} = 1 \tag{12}$$

The quaternion product is defined for quaternion operation.

$$\boldsymbol{q} \otimes \boldsymbol{p} = \begin{bmatrix} q_4 \bar{\boldsymbol{p}} + p_4 \bar{\boldsymbol{q}} - \bar{\boldsymbol{q}} \times \bar{\boldsymbol{p}} \\ q_4 p_4 - \bar{\boldsymbol{q}}^T \bar{\boldsymbol{p}} \end{bmatrix} \tag{13}$$

Equation (13) means rotating the attitude represented by a quaternion $\boldsymbol{p}$ by a quaternion $\boldsymbol{q}$. The kinematics of the quaternion $\boldsymbol{q}$ is wrriten as

$$\frac{\mathrm{d}}{\mathrm{d}t} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \frac{1}{2} \tilde{\omega} \otimes \boldsymbol{q} \tag{14}$$

$$= \frac{1}{2} \begin{bmatrix} 0 & \omega_z & -\omega_y & \omega_x \\ -\omega_z & 0 & \omega_x & \omega_y \\ \omega_y & -\omega_x & 0 & \omega_z \\ -\omega_x & -\omega_y & -\omega_z & 0 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \tag{15}$$

where $\tilde{\omega} = [\omega^T, 0]^T$

This paper uses the generalized Rodrigues parameters in the UKF, which are defined as

$$\boldsymbol{p} = f \frac{\bar{\boldsymbol{q}}}{a + q_4} \tag{16}$$

where a is a parameter from 0 to 1 and f is a scale factor.

Euler angles are the most famous definition of representing the orientation of an object. This paper uses them to show the results of numerical simulations. They can represent the attitude by conducting three consecutive rotations around each coordinate axis and can be understood intuitively, but they have singularities as a drawback. The rotation matrices around each axis of a frame are written as $R_z(\phi)$, $R_y(\theta)$, and $R_x(\psi)$ where ϕ, θ, and ψ are rotation angles of each axis, respectively. A rotation matrix from the inertial frame of reference (i-frame) to the body-fixed frame (b-frame) is described as

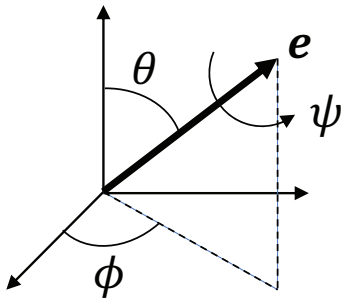


Fig. 3: Rotation axis unit vector

$$R_{b/i} = R_x(\psi)R_y(\theta)R_z(\phi) \tag{17}$$

$$= \begin{bmatrix} \cos\theta\cos\phi & \cos\theta\sin\phi & -\sin\theta \\ \sin\theta\sin\psi\cos\phi - \cos\psi\sin\phi & \sin\theta\sin\psi\sin\phi + \cos\psi\cos\phi & \cos\theta\sin\psi \\ \sin\theta\cos\psi\cos\phi + \sin\psi\sin\phi & \sin\theta\cos\psi\sin\phi - \sin\psi\cos\phi & \cos\theta\cos\psi \end{bmatrix} \tag{18}$$

Furthermore, an attitude can be represented by the unit vector of the rotation axis $\boldsymbol{e}$ and the rotation angle ψ shown in Fig. 3. The relational expression between the rotation vector and the quaternions is written as

$$\boldsymbol{q} = \begin{bmatrix} \sin\theta\cos\phi\sin\frac{\psi}{2} \\ \sin\theta\sin\phi\sin\frac{\psi}{2} \\ \cos\theta\sin\frac{\psi}{2} \\ \cos\frac{\psi}{2} \end{bmatrix} \tag{19}$$

2.4 Dynamics of a Space Object

A space object is assumed to be a rigid body, and Euler's equation is used to represent the object's attitude motion. It follows the conservation of angular momentum as

$$\frac{d\boldsymbol{h}}{dt} = \frac{d^*\boldsymbol{h}}{dt} + \boldsymbol{\omega}\times\boldsymbol{h} \tag{20}$$

$$= J\frac{d^*\boldsymbol{\omega}}{dt} + \boldsymbol{\omega}\times J\boldsymbol{\omega} = \boldsymbol{\tau} \tag{21}$$

where $\boldsymbol{h}$ is angular momentum vector, J is the moment of inertia, $\boldsymbol{\tau}$ is external torque, and $\frac{d^*}{dt}$ is the time differential with respect to the body-fixed frame.

2.5 Gaussian Process Regression (GPR)

Gaussian process regression (GPR) is one of the machine learning methods [7]. The regression model of the function $y = f(\boldsymbol{x})$ has the scalar output y from the input vector $\boldsymbol{x}$ with dimension $m\times 1$. A training data set is represented as

$$\mathcal{D} = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_N, y_N)\} = \langle X, \boldsymbol{y}\rangle \tag{22}$$

where $\boldsymbol{x}_k$ $(k = 1, \ldots, N)$ is an input vector and y_k is the scalar output of the corresponding input $\boldsymbol{x}_k$. The output vector $\boldsymbol{y} = [y_1, y_2, \ldots, y_N]^T$ is the multivariate Gaussian distribution normalized to have a mean of zero and its covariance matrix K as

$$\boldsymbol{y} \sim \mathcal{N}(0, K) \tag{23}$$

where K is the $N\times N$ kernel matrix of the input data. The element $K_{nn'}$ is defined as

$$K_{nn'} = k(\boldsymbol{x}_n, \boldsymbol{x}_{n'}) + \epsilon^2 I \tag{24}$$

Algorithm 1 The GPR algorithm

$[mu, var] = gpr(xtest, xtrain, ytrain, kernel)$
$N = length(ytrain)$
for $n = 1, \ldots, N$ **do**
  for $n' = 1, \ldots, N$ **do**
    $K[n, n'] = kernel(xtrain[n], xtrain[n'])$
  end for
end for
$yy = K^{-1} * ytrain$
for $m = 1, \ldots, M$ **do**
  for $n = 1, \ldots, N$ **do**
    $k[n] = kernel(xtrain[n], xtest[m])$
  end for
  $s = kernel(xtest[m], xtest[m])$
  $mu[m] = k^T * yy$
  $var[m] = s - k^T * K^{-1} * k$
end for

where $k(\cdot)$ is the kernel function, ϵ^2 is the variance of the observation noise, and I is the identity matrix. In this study, the Gaussian kernel represented by Eq. (25) is used.

$$k(\boldsymbol{x}, \boldsymbol{x}') = \theta_1 \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{\theta_2}\right) \tag{25}$$

where θ_1 and θ_2 are hyperparameters of the GPR.

Given a new input vector $\boldsymbol{x}^*$ from the training dataset $\mathcal{D}$, the predictive distribution of y^* is represented as

$$p(y^*|\boldsymbol{x}^*, \mathcal{D}) = \mathcal{N}(\boldsymbol{k}_*^T K^{-1} \boldsymbol{y}, k_{**} - \boldsymbol{k}_*^T K^{-1} \boldsymbol{k}_*) \tag{26}$$

That is, the output y^* corresponding to the input $\boldsymbol{x}^*$ can be obtained from the Gaussian process trained by data set $\mathcal{D}$. The mean and the variance of y^* regressed by GP are respectively denoted as

$$\mathrm{GP}_\mu(\boldsymbol{x}^*, \mathcal{D}) = \boldsymbol{k}_*^T K^{-1} \boldsymbol{k}_* \tag{27}$$

$$\mathrm{GP}_\Sigma(\boldsymbol{x}^*, \mathcal{D}) = k_{**} - \boldsymbol{k}_*^T K^{-1} \boldsymbol{k}_* \tag{28}$$

where

$$\boldsymbol{k}_* = [k(\boldsymbol{x}^*, \boldsymbol{x}_1), k(\boldsymbol{x}^*, \boldsymbol{x}_2), \ldots, k(\boldsymbol{x}^*, \boldsymbol{x}_N)]^T \tag{29}$$

$$k_{**} = k(\boldsymbol{x}^*, \boldsymbol{x}^*) \tag{30}$$

The algorithm of GPR is summarized in Algorithm 1.

2.6 Unscented Kalman Filter (UKF)

A discrete-time nonlinear system model is given by

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(k, \boldsymbol{x}_k, \boldsymbol{v}_k, \boldsymbol{u}_k) \in \mathbb{R}^m \tag{31}$$

$$\boldsymbol{y}_k = \boldsymbol{h}(k, \boldsymbol{x}_k, \boldsymbol{\epsilon}_k, \boldsymbol{u}_k) \in \mathbb{R}^l \tag{32}$$

where $\boldsymbol{x}_k$ is the $m \times 1$ state vector, $\boldsymbol{u}_k$ is the input vector, $\boldsymbol{y}_k$ is the $l \times 1$ observation vector at discrete-time t_k, and $\boldsymbol{v}_k$ and $\boldsymbol{\epsilon}_k$ are the process noise and the observation noise, respectively. The UKF approximates a Gaussian distribution using sigma points that are obtained as

$$\boldsymbol{\chi}_{k-1,0} = \hat{\boldsymbol{x}}_{k-1} \tag{33}$$

$$\boldsymbol{\chi}_{k-1,i} = \boldsymbol{\chi}_{k-1,0} + \boldsymbol{\sigma}_{k-1,i} \tag{34}$$

$$\boldsymbol{\sigma}_{k-1} = [\sqrt{m+\lambda}\sqrt{P_{k-1}}, -\sqrt{m+\lambda}\sqrt{P_{k-1}}] \in \mathbb{R}^{m \times 2m} \tag{35}$$

where $\hat{\boldsymbol{x}}_{k-1}$ is the state estimate at t_{k-1}, $P_k = \sqrt{P_k}\sqrt{P_k}^T$ is the $m \times m$ covariance matrix of the $\boldsymbol{x}_k$. Scaling parameter λ is defined as

$$\lambda = \alpha^2 (m + \kappa) - m \tag{36}$$

where $\kappa = 3 - m$ and $10^{-4} < \alpha < 1$ are tuning parameters. The substitution of sigma points $\boldsymbol{\chi}_k$ into Eq. (31) yields predicted sigma points $\boldsymbol{\chi}_k^-$

$$\boldsymbol{\chi}_k^- = \boldsymbol{f}(k-1, \boldsymbol{\chi}_{k-1}, \boldsymbol{u}_{k-1}) \tag{37}$$

Taking a weighted average of the predicted sigma points $\boldsymbol{\chi}_k^-$, a priori predicted state value and covariance matrix are

$$\hat{\boldsymbol{x}}_k^- = \sum_{i=0}^{2m} w_{i,\text{mean}} \boldsymbol{\chi}_{k,i}^- \tag{38}$$

$$P_k^- = \sum_{i=0}^{2m} w_{i,\text{cov}} \left(\boldsymbol{\chi}_{k,i}^- - \hat{\boldsymbol{x}}_k^-\right)\left(\boldsymbol{\chi}_{k,i}^- - \hat{\boldsymbol{x}}_k^-\right)^T \tag{39}$$

where weights $w_{i,\text{mean}}$ and $w_{i,\text{cov}}$ are given by

$$w_{0,\text{mean}} = \frac{\lambda}{n+\lambda} \tag{40}$$

$$w_{0,\text{cov}} = \frac{\lambda}{n+\lambda} + \left(1 - \alpha^2 + \beta\right) \tag{41}$$

$$w_{i,\text{mean}} = w_{i,\text{cov}} = \frac{1}{2(n+\lambda)}, \quad (i = 1, 2, \ldots, 2m) \tag{42}$$

Sigma points $\hat{\boldsymbol{\chi}}_k^-$ in Eq. (38) are calculated again and used for the calculation of a priori predicted observation $\hat{\boldsymbol{y}}_k^-$ as

$$\boldsymbol{\mathcal{Y}}_{k,i}^- = \boldsymbol{h}(\boldsymbol{\chi}_{k,i}^-), \quad (i = 0, 1, \cdots, 2n) \tag{43}$$

$$\hat{\boldsymbol{y}}_k^- = \sum_{i=0}^{2m} w_{i,\text{mean}} \boldsymbol{\mathcal{Y}}_{k,i}^- \tag{44}$$

Kalman gain is given by

$$G_k = P_k^{xy} \left(P_k^{yy}\right)^{-1} \tag{45}$$

where covariance matrices P_k^{xy} and P_k^{yy} are defined as

$$P_k^{yy} = \sum_{i=0}^{2m} w_{i,\text{cov}} \left(\boldsymbol{\mathcal{Y}}_{k,i}^- - \hat{\boldsymbol{y}}^-\right)\left(\boldsymbol{\mathcal{Y}}_{k,i}^- - \hat{\boldsymbol{y}}^-\right)^T \in \mathbb{R}^{l \times l} \tag{46}$$

$$P_k^{xy} = \sum_{i=0}^{2m} w_{i,\text{cov}} \left(\boldsymbol{\chi}_k^{i,-} - \hat{\boldsymbol{x}}^-\right)\left(\boldsymbol{\mathcal{Y}}_{k,i}^- - \hat{\boldsymbol{y}}^-\right)^T \in \mathbb{R}^{m \times l} \tag{47}$$

Consequently, posterior estimated state value $\hat{\boldsymbol{x}}_k^+$ and covariance matrix P_k^+ are obtained as follows.

$$\hat{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^- + G_k\left(\tilde{\boldsymbol{y}}_k - \hat{\boldsymbol{y}}_k^-\right) \tag{48}$$

$$P_k^+ = P_k^- - G_k P_k^{yy} G_k^T \tag{49}$$

where $\tilde{y}$ is the vector of observed values at discrete-time t_k.

3. APPLYING GPUKF TO ATTITUDE ESTIMATION VIA LIGHT CURVES

Bayesian filters, such as the extended Kalman filter and UKF, are often used for the estimation of attitude via light curves. They need a state space model of the target object constructed by a prediction model and an observation model shown in Eqs. (50) and (51) respectively. However, the state space model cannot be constructed when some parameters are unknown. The observation model especially cannot be described if the shape or surface parameter is unknown. To tackle this problem, this paper proposes a state space model using GPR. The GPR enables the construction of the non-parametric model for estimation. The following describes how the GP model is combined with the UKF, called GPUKF.

3.1 Prediction Model and Observation Model

The prediction model calculates the attitude of the next time step from the current state vector. The observation model calculates the light curve from the current state vector. Each model is defined as

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_{\mathrm{prd},k}) \tag{50}$$

$$m_{\mathrm{app},k} = \mathrm{GP}(\boldsymbol{x}_{\mathrm{obs},k}, \mathcal{D}_{\mathrm{obs}}) \tag{51}$$

where $\boldsymbol{f}$ is the function of the dynamics of the object, $\boldsymbol{x}_{\mathrm{prd}}$ is the input vector for the prediction model, and $\boldsymbol{x}_{\mathrm{obs}}$ is the input vector for the observation model. This study uses Euler's equation in Eq. (21) as the prediction model $\boldsymbol{f}$ because the object–observer–Sun position is assumed to be fixed. This assumption is appropriate because the target object is on a geosynchronous orbit and the observation is for several minutes. The training data set $\mathcal{D}_{\mathrm{obs}}$ is created by the pairs of the input values $\boldsymbol{x}_{\mathrm{obs},k}$ and the output value $m_{\mathrm{app},k}$. How to create the training data set is described in Subsection 3.2.

3.2 Training Data

The training data set for the regression of the light curves is written as

$$\mathcal{D}_{\mathrm{obs}} = \langle X_{\mathrm{obs}}, Z \rangle \tag{52}$$

where X_{obs} is the set of inputs and Z is the set of outputs. They are defined as

$$X_{\mathrm{obs}} = [\boldsymbol{x}_{\mathrm{obs},1}, \boldsymbol{x}_{\mathrm{obs},2}, \ldots, \boldsymbol{x}_{\mathrm{obs},N}] \tag{53}$$

$$Z = [m_{\mathrm{app},1}, m_{\mathrm{app},2}, \ldots, m_{\mathrm{app},N}] \tag{54}$$

where the input vector $\boldsymbol{x}_{\mathrm{obs},k}$ in this paper is

$$\boldsymbol{x}_{\mathrm{obs},k} = [\boldsymbol{q}_k^T, m_{\mathrm{app,past},k}]^T \tag{55}$$

where $\boldsymbol{q}_k$ is the quaternion of the object and $m_{\mathrm{app,past},k}$ denotes the mean of the past light curves over several seconds. Equations (15), (21), and (8) are calculated to obtain $\boldsymbol{q}_k$ and $m_{\mathrm{app,past},k}$. The state space of attitude for training data needs to be covered entirely because the light curve corresponding to each attitude angle needs to be trained for accurate regression. Such training data can be obtained by varying the unit vector of the rotation axis in all directions shown in Fig. 4 and varying a rotation angle ψ for each rotation axis. Furthermore, each attitude is used as the initial state of Eqs. (15) and (21). In addition to attitude, the initial angular rate $\boldsymbol{\omega_0}$ is also set for propagation. Thus, the time history of the attitude for several seconds is obtained, and then the light curve corresponding to each attitude is calculated by Eq. (8). The final attitude and light curve in the time history are denoted as $\boldsymbol{q}_k$ and $m_{\mathrm{app},k}$.

In addition, the average of the past light curves for several seconds is denoted as $m_{\mathrm{app,past},k}$. By including $m_{\mathrm{app,past},k}$ in the input data of $\mathcal{D}_o$, the change of $m_{\mathrm{app},k}$ by the difference in the surface properties can be trained. The mean of the past light curves for several seconds can mitigate the influence of observation noise and train the observation model depending on the surface properties. Therefore, the observation model in Eq. (51) is expected to be robust against the uncertainty of the surface properties.

3.3 GPUKF Model for Attitude Estimation via Light Curves

The algorithm of the GPUKF model for attitude estimation is shown in Fig. 5. First, the estimated initial state values are substituted to the prediction model in Eq. (50) using sigma points in Eqs. (33)–(35) and the predicted state values are obtained as sigma points. Although (global) attitude is represented with quaternions, the (local) attitude error is

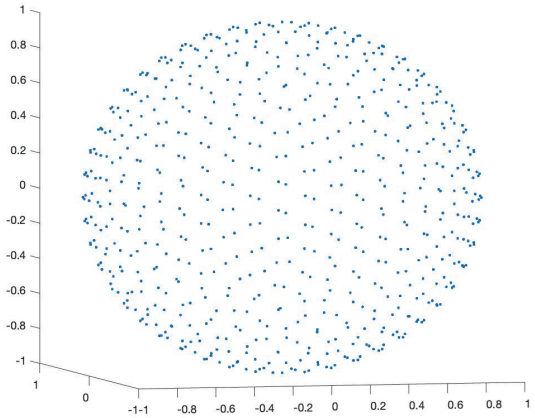


Fig. 4: Rotation axis unit vectors for covering the state space of attitude entirely

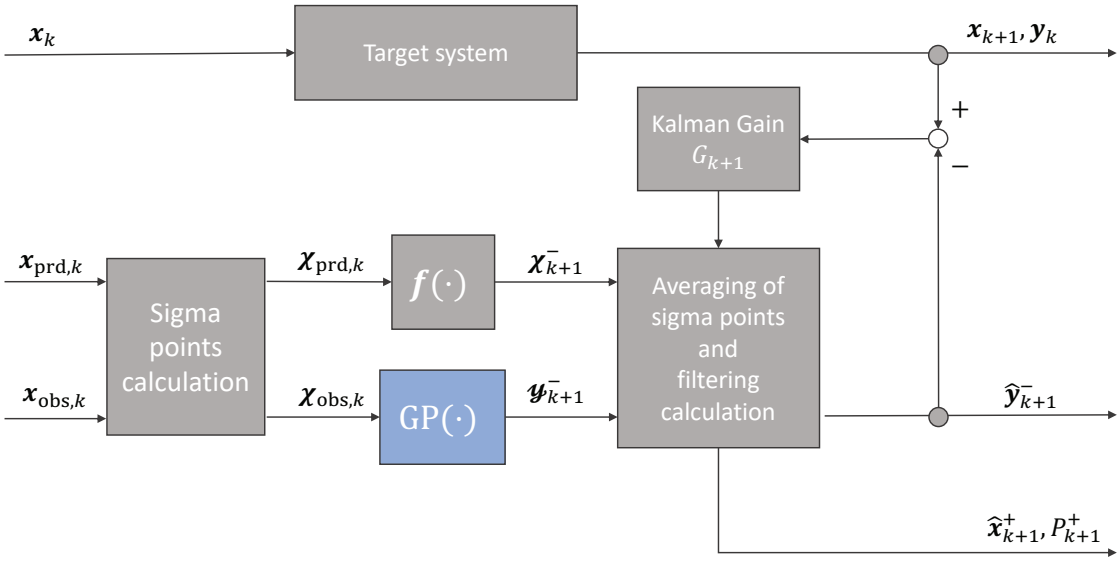


Fig. 5: Algorithm of GPUKF

described with the generalized GRP as proposed in [10]. Thus the sigma points are calculated for the GRP. Then, the sigma points of the predicted observed value in Eq. (43) are calculated by substituting the sigma points of the predicted state to the observation model of the GP in Eq. (51). Finally, the Kalman gain in Eq. (45) is calculated and the posterior estimated state value $\hat{\boldsymbol{x}}_k^+$ and the posterior covariance matrix P_k^+ are obtained by Eqs. (48) and (49). Additionally, $m_{\mathrm{app,past},k}$, one of the input values for GPUKF, is also updated by

$$m_{\mathrm{app,past},k+1} = (t_{\mathrm{span,past}} m_{\mathrm{app,past},k} + m_{\mathrm{app},k})/(t_{\mathrm{span,past}} + 1) \tag{56}$$

where $m_{\mathrm{app},k}$ is the observed value and $t_{\mathrm{span,past}}$ is the time period for the averaging of the past light curves.

4. NUMERICAL SIMULATION

4.1 Parameters for Training and Validation Data

Numerical simulations are conducted with the object–observer–Sun position shown in Table 1. It can be assumed to be fixed because the object is on a geosynchronous orbit and the observation is for several minutes. The shape of the

Table 1: Object-Observer-Sun fixed position (inertial frame)

Parameter	Value
Object position	$(0, 0, 0)$ km
Observer position	$(36000, 0, 0)$ km
Sun position	$(1.2975, -0.3461, -0.6592) \times 10^8$ km

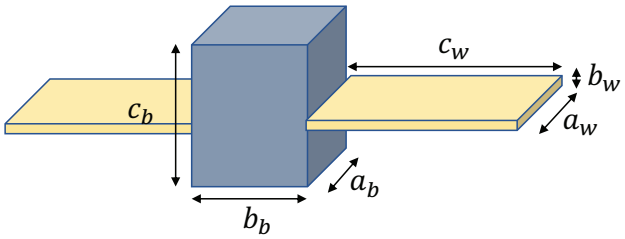


Fig. 6: Box-Wing

target object used in both the training and validation simulations is the box-wing satellite illustrated in Fig. 6 and its shape parameters are summarized in Table 2.

This study proposes an estimation method that has robustness against surface parameters by using GPUKF. As an example, the results of the attitude estimation with a conventional UKF are shown in Fig. 7. These results show the errors between the true attitude and the estimated attitude. The result shown in Fig. 7a is obtained when the surface parameters are the same between the target object (i.e., true values) and the UKF. The result shown in Fig. 7b is obtained when one of the surface parameters ρ is different between the target object and the UKF. As can be seen from these results, if ρ differs between the target object (i.e., true values) and the parameter used in UKF, the estimation error does not converge to zero. Therefore, the training data set $\mathcal{D}_{\text{obs}}$ for the observation model is created for various surface parameters so that robustness against the surface properties is achieved in the estimation of the attitude. The surface parameters of the Ashikhmin–Shirley model for the training data are set in Table 3, where the diffuse factor ρ has various values.

The inputs and outputs of the training data set in Eq. (52) are described in Subsection 3.2. In the following simulations, the initial attitudes are set to cover the state space entirely using the rotation axis unit vectors shown in Fig. 4 and applying rotation angles ψ in increments of 60 deg from 0 to 360 deg. The initial angular rate for propagation to create training data is $\boldsymbol{\omega}_0 = [0.1, -0.1, -0.25]^T$ deg/s. The calculations in Eqs. (15) and (21) are propagated numerically in increments of 1 s from 0 to 10 s. In addition, the time period for the averaging of the past light curves is set to $t_{\text{span,past}} = 10$ s.

The initial conditions for propagating the validation data, that is, the true attitude, are $\boldsymbol{q}_0 = [0, 0, 0, 1]^T$ and $\boldsymbol{\omega}_0 = [0.1, -0.1, -0.25]^T$ deg/s. The surface parameters for the validation are the same as those in Table 3, but the diffuse reflectance ρ is changed for each simulation.

4.2 Results

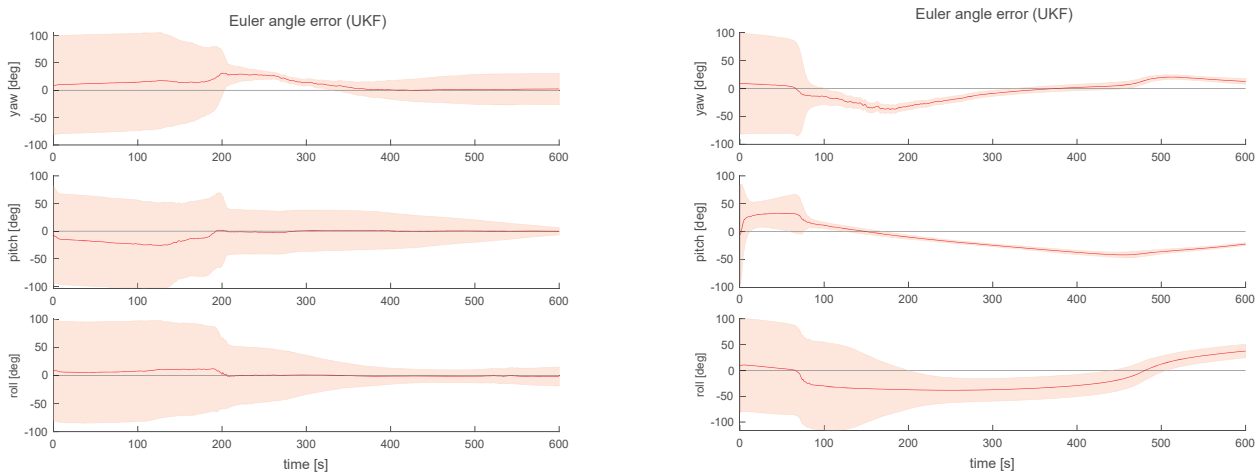
Figures 8–10 show the attitude estimation results represented with the Euler angle and the predicted light curves compared between GPUKF and UKF. The blue line is GPUKF, the red line is UKF, and the black line is the true values in both (a) and (b). The green line in (b) means the light curves calculated using the estimated attitude by GPUKF using Eq. (8). In these simulations, the estimated initial state values have errors $(\phi, \theta, \psi) = (-10, 5, -10)$ deg compared to the true value.

Table 2: Shape Parameters

Parameter	Value
Box side length, (a_b, b_b, c_b)	$(1.0, 1.0, 1.3)$ m
Wing side length, (a_w, b_w, c_w)	$(1.0, 0.1, 2.0)$ m
Moment of inertia , (J_x, J_y, J_z)	$(55, 40, 65)$ kgm^2

Table 3: Surface Parameters

Parameter	Value
Reflection properties, (n_u, n_v)	$(800, 800)$
Fresnel reflection, F_0	0.5
Fraction of the diffusion, d	1
Fraction of the specularity, s	1
Diffuse reflectance of the substrate, ρ	0.1, 0.5, 0.6, 0.75, 0.9



(a) $\rho_{\text{ukf}} = \rho_{\text{true}} = 0.5$ **(b)** $\rho_{\text{ukf}} = 0.5, \rho_{\text{true}} = 0.7$

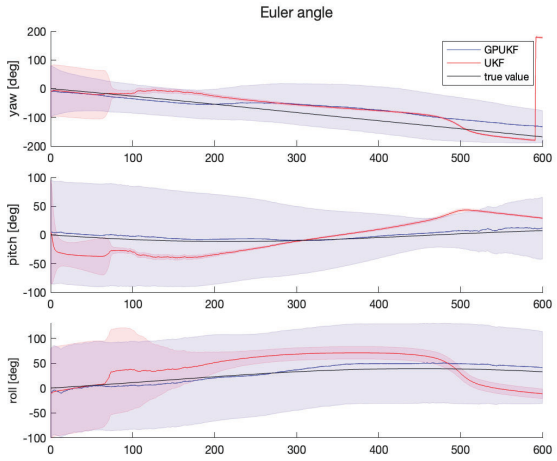
Fig. 7: The attitude estimation with UKF is sensitive to the difference between ρ_{true} and ρ_{ukf}

Figures 8 and 9 show that GPUKF successfully estimates the attitude of the object with various ρ_{true}, although UKF does not. In other words, the attitude estimation with GPUKF is more robust against the uncertainty of the target object's surface property than UKF. The predicted light curves of GPUKF in Figs. 8(b) and 9(b) are much more accurate than those of UKF. This means that the observation model with GP in Eq. (51) learns the difference of the object's surface property.

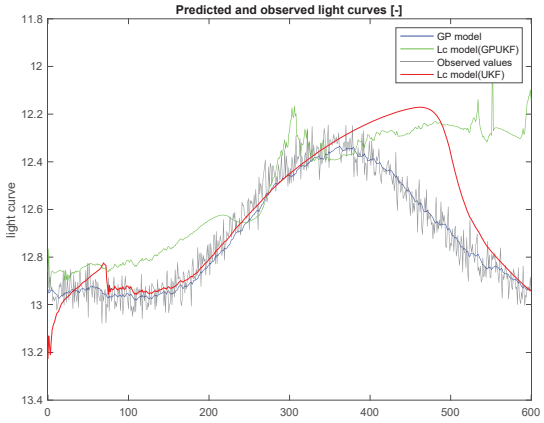
Figure 10 represents the attitude estimation without $m_{\text{app,past}}$ in the input data of GPUKF. If $m_{\text{app,past}}$ is not included in the input data, the attitude estimation with GPUKF fails because the difference of the target's surface property cannot be identified in the regression.

5. CONCLUSION

This paper addresses a non-parametric model of light curves using Gaussian process regression (GPR). When the attitude estimation via light curves is conducted for the RSO with unknown surface properties, the conventional estimation scheme suffers from the difficulty that the observation model cannot be constructed parametrically because the light curves depend on surface properties. This study proposes a method of describing the observation model using GP and combines it with UKF (GPUKF) for the estimation of the attitude using light curves. The observation model for UKF can be written non-parametrically using GP, and the observation model can be constructed for the RSO with unknown surface properties. The training data set of GPUKF is created for various surface parameters so that the robustness to the uncertainty of the target object's surface property is achieved. The results of the numerical simulations show that the attitude can be estimated via light curves using GPUKF. Furthermore, the comparison between GPUKF and UKF shows that the GPUKF can regress the light curves under the condition that the UKF cannot. Thus, GPUKF achieves robustness against the uncertainty of the surface property. The attitude estimation of the non-resolved object via light curves can be more developed by refining this novel method, contributing to practical scenarios in SSA/SDA where object parameters are unknown.

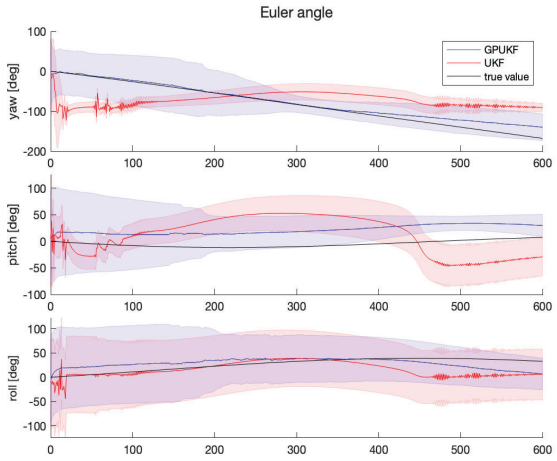


(a) Euler angle estimation

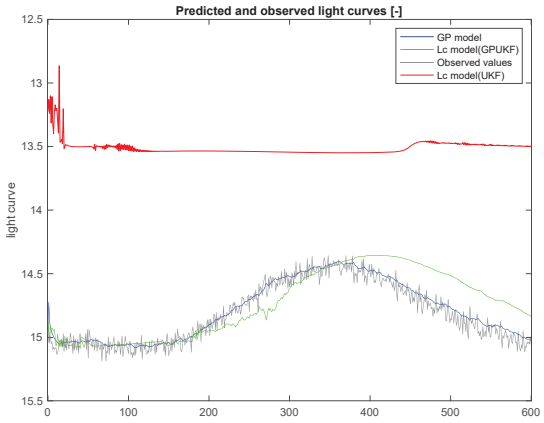


(b) Predicted light curves

Fig. 8: The comparison of the attitude estimation between the attitude estimation with GPUKF and the one with UKF ($\rho_{\text{ukf}} = 0.5, \rho_{\text{true}} = 0.7$)

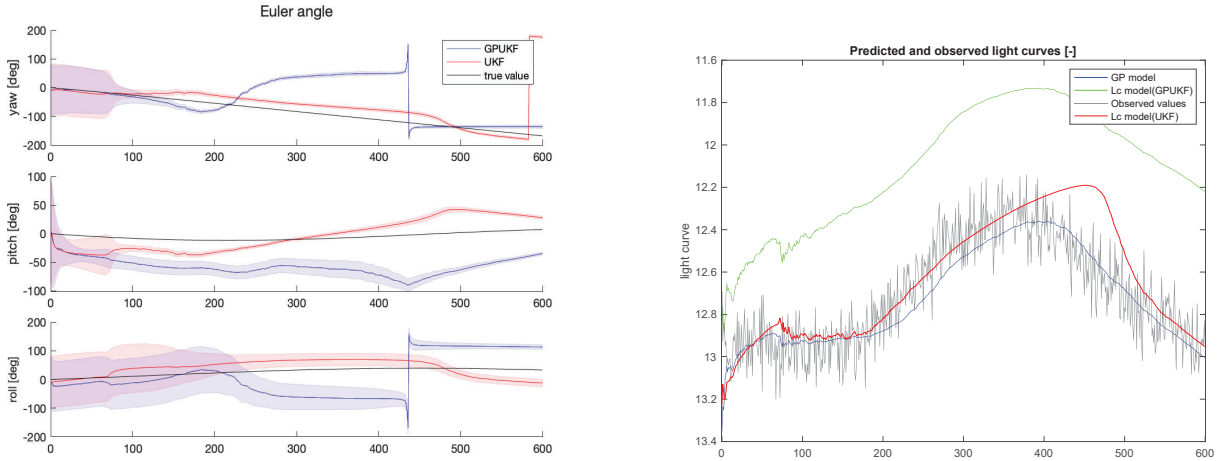


(a) Euler angle estimation



(b) Predicted light curves

Fig. 9: The comparison of the attitude estimation between the attitude estimation with GPUKF and the one with UKF ($\rho_{\text{ukf}} = 0.5, \rho_{\text{true}} = 0.1$)



(a) Euler angle estimation

(b) Predicted light curves

Fig. 10: The comparison of the attitude estimation between the attitude estimation with GPUKF not using $m_{\text{app,past}}$ as one of the inputs and the one with UKF ($\rho_{\text{ukf}} = 0.5, \rho_{\text{true}} = 0.7$)

REFERENCES

[1] Heather Cowardin. Orbital debris: Quarterly news. *Orbital Debris Quarterly News*, 26(1), 2022.

[2] K Muinonen, J Torppa, X-B Wang, A Cellino, and A Penttilä. Asteroid lightcurve inversion with bayesian inference. *Astronomy & Astrophysics*, 642:A138, 2020.

[3] TG Müller, J Ďurech, Masateru Ishiguro, Michael Mueller, T Krühler, Hao Yang, M-J Kim, Laurence O'Rourke, Fumihiko Usui, Casba Kiss, et al. Hayabusa-2 mission target asteroid 162173 ryugu (1999 ju3): Searching for the object's spin-axis orientation. *Astronomy & Astrophysics*, 599:A103, 2017.

[4] Fabrizio Piergentili, Fabio Santoni, and Patrick Seitzer. Attitude determination of orbiting objects from lightcurve measurements. *IEEE Transactions on Aerospace and Electronic Systems*, 53(1):81–90, 2017.

[5] Charles J Wetterer and Moriba Jah. Attitude determination from light curves. *Journal of Guidance, Control, and Dynamics*, 32(5):1648–1651, 2009.

[6] Yuri Matsushita, Yasuhiro Yoshimura, Shuji Nagasaki, and Toshiya Hanada. Conceptual study of improved photometric attitude estimation using glint. *TRANSACTIONS OF THE JAPAN SOCIETY FOR AERONAUTICAL AND SPACE SCIENCES, AEROSPACE TECHNOLOGY JAPAN*, 22:59–65, 05 2024.

[7] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian processes for machine learning*. Adaptive computation and machine learning. MIT Press, 2006.

[8] Daniël Jimenez Kwast. An introduction to brdf models. *Hmi. Ewi. Utwente, NI*, 2014.

[9] Michael Ashikhmin and Peter Shirley. An anisotropic phong brdf model. *Journal of graphics tools*, 5(2):25–32, 2000.

[10] John L Crassidis and F Landis Markley. Unscented filtering for spacecraft attitude estimation. *Journal of guidance, control, and dynamics*, 26(4):536–542, 2003.